

# Introduction to computers and computing environments

Mark Stenglein, GDW 2019 pre-workshop crash course  
[StengleinLab.org](http://StengleinLab.org)



Math  
undergrad

7 years as a  
software engineer

PhD in mol.  
biology /  
biochem.

Postdoc using  
microarrays, NGS,  
and bioinformatics

Asst. Professor  
at CSU

1999, Bangkok, Thai Airways test facility



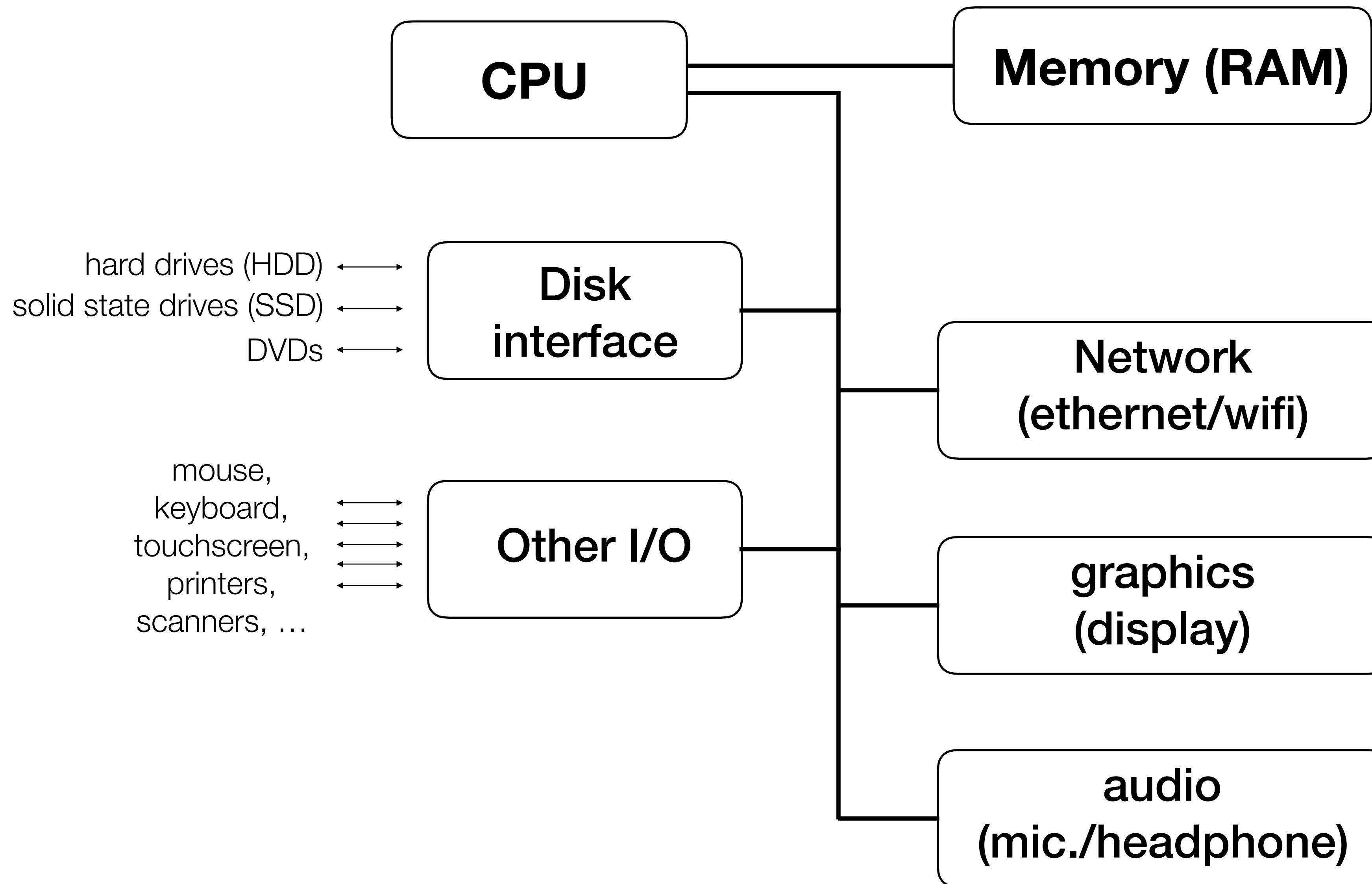
The Arthropod-Borne and  
Infectious Diseases Laboratory  
3185 Rampart Road Building T

AIDL  
Colorado State  
University

Director, CSU  
NGS facility



# Basic computer architecture



# Memory (RAM) vs. storage (drives)

	<b>Hard drives (HDD)</b>	<b>Solid state drives (SSD)</b>	<b>Main memory (RAM)</b>
Speed	Slow	Much faster	Fastest
Cost per Gb	\$0.025	\$0.20	\$4
Volatile	No	No	Yes
Uses	Long term storage of large datasets	Laptops, long term storage, OS boot drives	
Other comments	Uses more power	Largest SSD ~4 Tb	Some bioinformatic tasks, like genome assembly, require lots of RAM



# How much RAM do you need?

We will be doing a number of typical bioinformatic tasks on your laptops this week, which have 16 Gb of RAM. That much RAM is plenty for many tasks.

Some bioinformatics tasks are processor (CPU) intensive, but use little RAM. For instance, inferring phylogenetic trees.

Other tasks, like assembly of large eukaryotic genomes or searching large databases with tools like BLAST, require lots of RAM. These types of analyses are better done on a dedicated server, a ‘super-computer’, or in the cloud.



# CPUs vs. cores vs. threads

Most modern computers have more than one CPU

Typically, these are referred to as “cores”.



To make it more confusing, Intel cores are “hyper-threaded”, which means that the operating systems sees these 4 cores as 8 CPUs.

```
[MDSTENGL-M01:Desktop _mdstengl$ sysctl hw.physicalcpu hw.logicalcpu
hw.physicalcpu: 4
hw.logicalcpu: 8
```

# Multi-threading

You can speed up your analyses by taking advantage of multiple CPUs/cores/threads.

Lots of times, an analysis will run quickly and you don't need to worry about this.  
Some computational tasks, like tree inference or genome assembly are computationally intensive.

## 1. Some programs have built-in multi-threading.

```
-num_threads <Integer, >=1>
Number of threads (CPUs) to use in the BLAST search
Default = '1'
```

### 1. If your computer has multiple processors/cores, use -p

The `-p` option causes Bowtie 2 to launch a specified number of parallel search threads. Each thread runs on a different processor/core and all threads find alignments in parallel, increasing alignment throughput by approximately a multiple of the number of threads (though in practice, speedup is somewhat worse than linear).

## 2. You can run multiple instances of single-threaded programs

1. You can do this manually
2. `parallel` is a linux command line utility for doing this
3. “Workflow managers” like Nextflow or snakemake allow parallelization.

# Speed-up from running an analysis in parallel

map 1M 50 nt reads to the human genome with 1 CPU: 1m 16 seconds

```
[mdstengl@cctsi-104:~/analyses/test_human_mapping$ time ./run_bowtie ERR3252925_1_1M.fastq GCA_000001405.15_GRCh38_no_alt_analysis_set.fna.bowtie_index
bowtie2 -x GCA_000001405.15_GRCh38_no_alt_analysis_set.fna.bowtie_index -q -U ERR3252925_1_1M.fastq --local --score-min C,1
00,1 --no-unal --threads 1 -S ERR3252925_1_1M.fastq.GCA_000001405.15_GRCh38_no_alt_analysis_set.fna.bowtie_index.sam

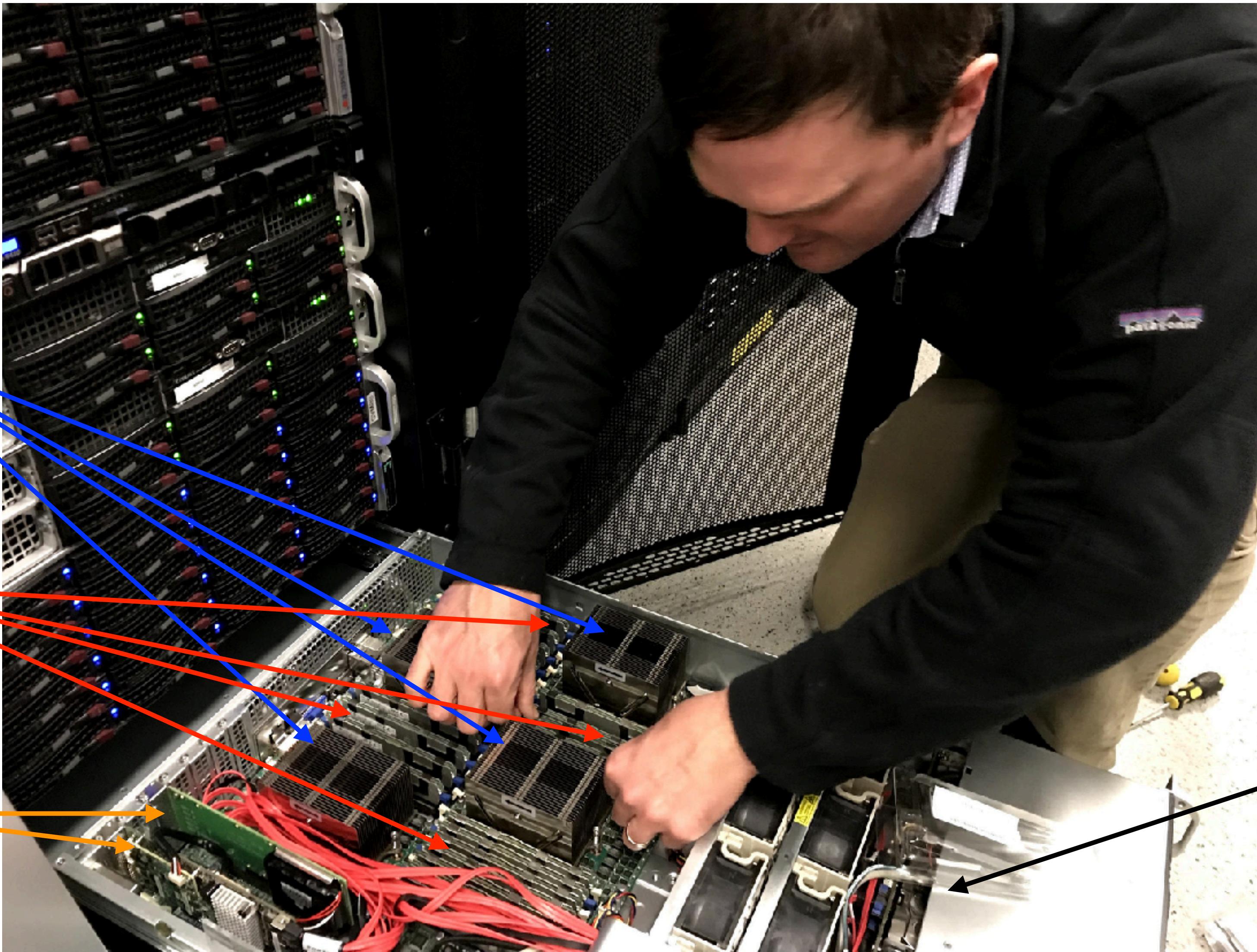
real    1m16.427s
user    1m13.836s
sys     0m12.844s
```

map 1M 50 nt reads to the human genome with 24 CPU: 9.6 seconds

```
[mdstengl@cctsi-104:~/analyses/test_human_mapping$ time ./run_bowtie_multiple_threads ERR3252925_1_1M.fastq GCA_000001405.15
_GRCh38_no_alt_analysis_set.fna.bowtie_index
bowtie2 -x GCA_000001405.15_GRCh38_no_alt_analysis_set.fna.bowtie_index -q -U ERR3252925_1_1M.fastq --local --score-min C,1
00,1 --no-unal --threads 24 -S ERR3252925_1_1M.fastq.GCA_000001405.15_GRCh38_no_alt_analysis_set.fna.bowtie_index.sam

real    0m9.641s
user    1m38.696s
sys     0m33.124s
```

# A bioinformatics server



CPUs  
 $(4 \times 16 \text{ cores} = 64)$

RAM  
 $(16 \times 32\text{Gb} = 512 \text{ Gb})$

ethernet  
(main user interface)

HDDs (6 x 8 Tb)  
& SSDs (2 x 128Gb)  
RAID configured

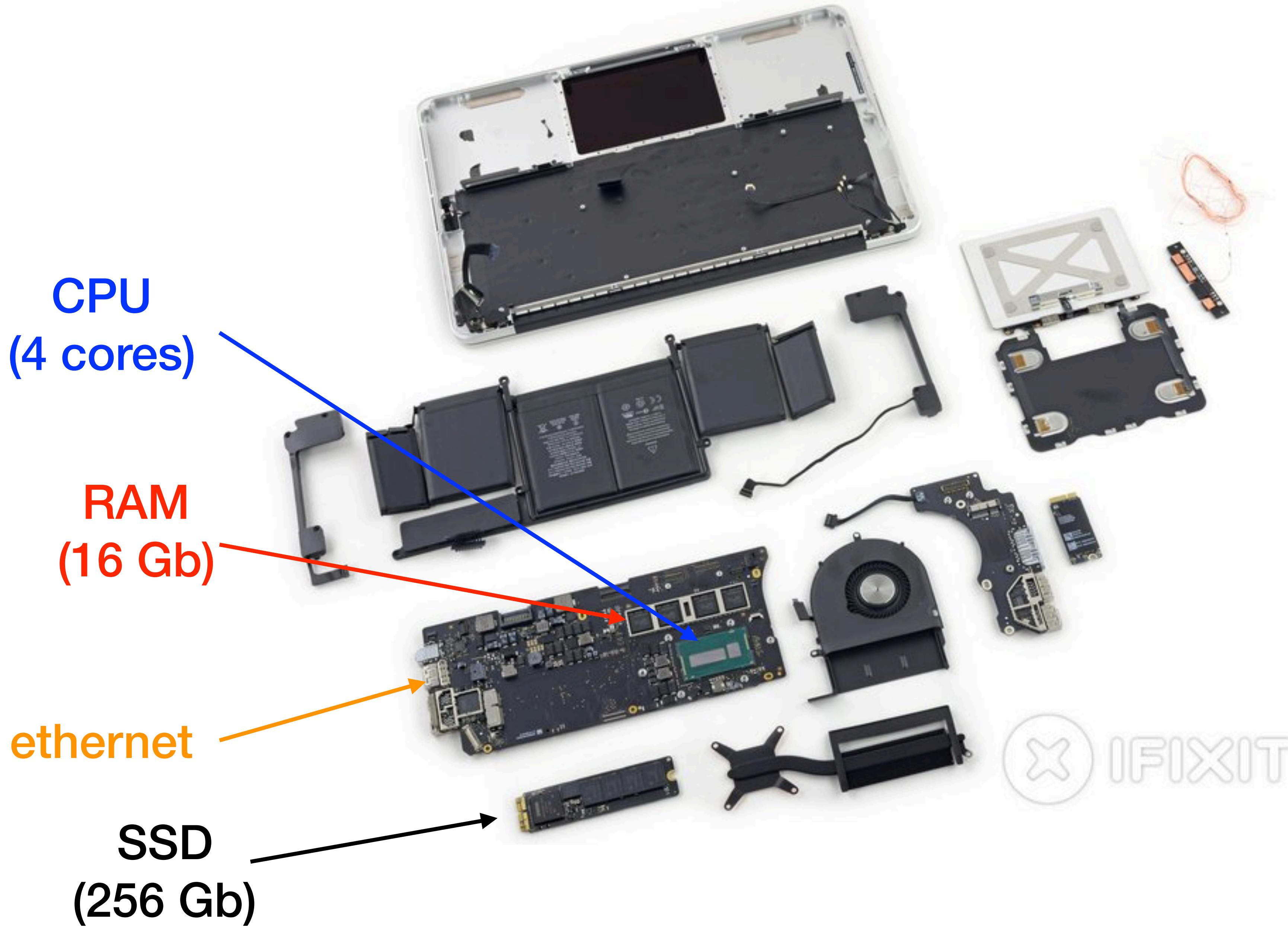
# How do you check resource usage and availability?

## **On your workshop laptops:**

- How much memory is available? How much is used?
- How much storage is available? What type is it? How much is used?
- How many cores and threads are there? What fraction of these are being used?



# The inside of your workshop laptops



# Checking resource usage and availability through the command line

command	better way to run it
df	reports disk usage and storage
top	reports CPU & memory usage

## Use these commands to answer the same questions:

- How much storage is available? How much is used?
- How much memory is available? How much is used?
- How many cores and threads are there? What fraction of these are being used?

# Local vs. remote computing

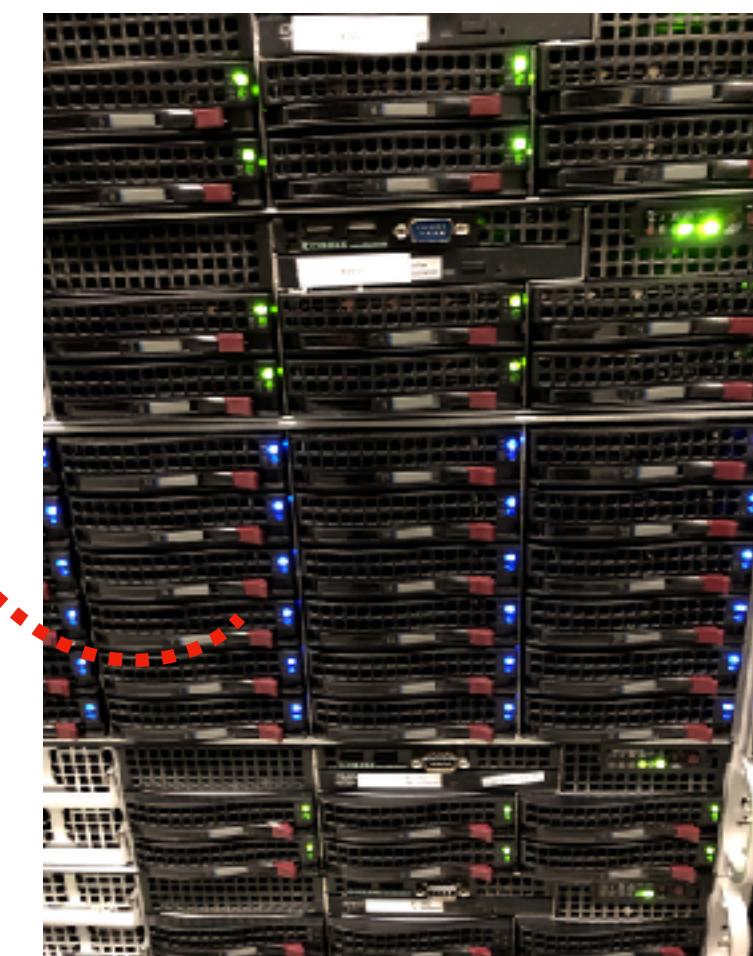
Local: you are using the resources  
on your computer



Remote: you are using the resources of a  
distant computer, but probably still  
connected through your own computer



a server  
somewhere



# Let's connect to a remote server!

**Connect to a bioinformatics server at CSU: Open the terminal app and run:**

```
ssh gdw@cctsi-104.cvmbs.colostate.edu
```

**Use the df and top commands to answer the same questions about this server**

- How much storage is available? How much is used? (hint: in /home)
- How much memory is available? How much is used?
- How many cores are there? What fraction of these are being used?

# Cloud computing is an increasingly popular form of remote computing

## Advantages:

- Scalable and flexible
- Don't have to buy or maintain servers
- Can take advantage of pre-existing images and containers

## Disadvantages:

- Can be expensive
- Have to pay to store and transfer data
- Can be slow to transfer data
- 

Real physical computing resources are available as virtual computers through the internet



image: gigabitmagazine.com

# Cost of cloud computing depends on how many resources you need

	vCPU	ECU	Memory (GiB)	Instance Storage (GB)	Linux/UNIX Usage
t2.large	2	Variable	8 GiB	EBS Only	\$0.1104 per Hour
t2.xlarge	4	Variable	16 GiB	EBS Only	\$0.2208 per Hour
r5.12xlarge	48	173	384 GiB	EBS Only	\$3.36 per Hour
r5.24xlarge	96	347	768 GiB	EBS Only	\$6.72 per Hour

Similar to the workshop laptops

Similar to the servers my lab uses

Plus ~\$0.05-\$0.15 per Gb-month for (short term) storage: \$50-\$150 per month per Tb

Plus ~\$0.09 per Gb to transfer from Amazon to anywhere else: \$90 per Tb

# An Amazon Web Services (AWS) linux environment

The screenshot shows the AWS EC2 console interface. On the left, the navigation menu is visible with sections like EC2 Dashboard, Instances, Images, Elastic Block Store, Network & Security, Load Balancing, Auto Scaling, and Systems Manager. The main area displays a terminal session on an Ubuntu 18.04.2 LTS instance (i-094d903b47d77bec). The terminal output includes system documentation links, system information (load, memory, swap), package updates, and a root login prompt. To the right of the terminal, there is a table showing two IPv6 addresses associated with the instance. At the bottom, detailed instance metadata is listed, including Private DNS, Private IP, Secondary private IPs, VPC ID, and Subnet ID.

EC2 Dashboard  
Events  
Tags  
Reports  
Limits  
INSTANCES  
Instances  
Launch Templates  
Spot Requests  
Reserved Instances  
Dedicated Hosts  
Scheduled Instances  
Capacity Reservations  
IMAGES  
AMIs  
Bundle Tasks  
ELASTIC BLOCK STORE  
Volumes  
Snapshots  
Lifecycle Manager  
NETWORK & SECURITY  
Security Groups  
Elastic IPs  
Placement Groups  
Key Pairs  
Network Interfaces  
LOAD BALANCING  
Load Balancers  
Target Groups  
AUTO SCALING  
Launch Configurations  
Auto Scaling Groups  
SYSTEMS MANAGER  
SERVICES

Launch Instance Connect

Filter by tags and attributes or search

Name Instance ID

i-01db0c57  
i-094d903b47d77bec

Downloads — ubuntu@ip-172-31-49-140: ~ — ssh -i mds\_linux.pem ubuntu@ec2-34-221-17-148.us-west-2.compute.amazonaws.com — 126x46

[MDSTENGL-M01:Downloads \_mdstengl\$ ssh -i "mds\_linux.pem" ubuntu@ec2-34-221-17-148.us-west-2.compute.amazonaws.com

Welcome to Ubuntu 18.04.2 LTS (GNU/Linux 4.15.0-1032-aws x86\_64)

\* Documentation: <https://help.ubuntu.com>  
\* Management: <https://landscape.canonical.com>  
\* Support: <https://ubuntu.com/advantage>

System information as of Wed May 29 16:56:27 UTC 2019

System load: 0.0 Processes: 84  
Usage of /: 13.7% of 7.69GB Users logged in: 0  
Memory usage: 14% IP address for eth0: 172.31.49.140  
Swap usage: 0%

Get cloud support with Ubuntu Advantage Cloud Guest:  
<http://www.ubuntu.com/business/services/cloud>

0 packages can be updated.  
0 updates are security updates.

Last login: Wed May 29 16:56:04 2019 from 129.82.26.66  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo\_root" for details.

ubuntu@ip-172-31-49-140:~\$

Instance: i-094d903b47d77bec

Description Status Checks

Instance ID: i-094d903b47d77bec  
Instance state: running  
Instance type: t2.micro  
Elastic IPs: None  
Availability zone: us-west-2a  
Security groups: launch-wizard-2, view inbound rules, view outbound rules  
Scheduled events: No scheduled events  
AMI ID: ubuntu/images/hvm-ssd/ubuntu-bionic-18.04-amd64-server-20190212.1 (ami-005bd005fb00e791)

Private DNS: ip-172-31-49-140.us-west-2.compute.internal  
Private IP: 172.31.49.140  
Secondary private IPs: None  
VPC ID: vpc-bfbb11c7  
Subnet ID: subnet-c7d3f28c

Feedback English (US)

© 2008 - 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use